On rare occasions, a granule (file) from Data Collection Version 2 (algorithm v2.8), may have all data fields of Null-length, because the granule is not populated due to lack of successfully retrieved soundings . Users can still open this HDF5 file, list contents and see all fields names, with no problems or alerting signs. However, the field names will not have a single byte of data behind them, i.e. the data objects will be of Null length.

The impact is that some HDF5 application interfaces (API), like IDL, will report an error, and will halt the program execution upon attempt to read such a Null-length field into allocated memory. IDL will issue an error message:

```
% Array dimensions must be greater than 0
```

and will halt the program execution, with an open API hanging in the air.

To avoid these interruptions, it is strongly recommended to apply a simple HDF5 query, before attempting to read data into memory. The query is to be placed after the API to the file and the data field are opened, and before attempting to read the data. Since this query is provided in the HDF5 API library, it would be a good programming practice to use it anyway in general, not only for ACOS data. In IDL, the name of this query function is H5D_GET_STORAGE_SIZE, and an example usage is shown below.

```
File_name ="acos_L2s_100424_41_Production_v100100_L2s2800_r01_PolB_110220040640.h5"

f_id=H5F_OPEN(File_name)
d_id=H5D_OPEN(f_id,'/RetrievalResults/xco2')

sz=H5D_GET_STORAGE_SIZE(d_id)  ;check the size of "xco2" data field

IF sz EQ 0 THEN BEGIN
         ;If size has been detected as 0, the execution detours through
         ;here, and the access (the API) to the file will close
         ;normally:

        H5D_CLOSE,d_id
        H5F_CLOSE,f_id

          ;Now you can skip to the next file, or gracefully
          ;exit the program, with a warning message if you wish.

ENDIF
```